

CLAIM AMENDMENTS

Claim Amendment Summary

Claims pending

- Before this Amendment: Claims 1-28.
- After this Amendment: Claims 1-32.

Non-Elected, Canceled, or Withdrawn claims: 28

Amended claims: 1-27.

New claims: 29-32

Claims:

1. **(Currently Amended)** At least one computer-readable storage medium having computer executable instructions that provide a method for transferring computer-readable objects across a remote boundary, the method comprising:

~~decomposing an object of a first type into an hierarchy of at least one sub-component based on a list of known object types, each sub-component comprising a known object associated with one of the known object types;~~

~~serializing the known objects into a serialized package; and~~
without relying on a pre-defined definition of a specified data type of an object, decomposing the object into multiple sub-components, wherein

the decomposing comprises extracting discernable properties and values for each sub-component;

without relying on the pre-defined definition of the specified data type of the object, serializing the multiple sub-components and their discernable properties and values into a serialized package; and

transmitting the serialized package to a remote entity, wherein the decomposing, serializing, and transmitting facilitates transferring computer-readable objects across a remote boundary.

2. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, wherein the list identifies the specified data type of the object ~~first type~~ as one of the known object types.

3. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, wherein at least one sub-component comprises an unknown object having a type unidentified within the list.

4. (Currently Amended) The computer-readable storage medium of claim 3, wherein decomposing an object further comprises decomposing the unknown object into another level of sub-components based on the list.

5. **(Currently Amended)** The computer-readable storage medium of claim 1, wherein a first process on a system transmits the serialized package and the remote entity comprises another process on the system.

6. **(Currently Amended)** The computer-readable storage medium of claim 1, wherein a first process on a system transmits the serialized package and the remote entity comprises another process on another system.

7. **(Currently Amended)** The computer-readable storage medium of claim 1, wherein a first application domain executing within a process transmits the serialized package and the remote entity comprises another application domain within the process.

8. **(Currently Amended)** The computer-readable storage medium of ~~claim 1~~ claim 29, wherein the hierarchy comprises a property bag.

9. **(Currently Amended)** The computer-readable storage medium of claim 8, wherein the property bag comprises a hash table.

10. (Currently Amended) The computer-readable storage medium of claim 9, wherein a key for each entry in the hash table comprises a name for the sub-component associated with the entry.

11. (Currently Amended) The computer-readable storage medium of claim 8, wherein the property bag comprises a plurality of entries, each entry being associated with one of the sub-components and having a first field for storing a name associated with the sub-component, a second field for storing a value associated with the sub-component, and a third field for storing a type associated with the sub-component.

12. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising negotiating the known object types identified within list by receiving a version number of a first list available to a first process, comparing the version number to another version number of a second list available to the remote entity, and determining the list based on the comparison.

13. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising negotiating the list by accepting a plurality of object types received from a first process, the accepted object types becoming known object types identified within the list.

14. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising negotiating the list by receiving an identifier for a file and having the list include object types identified within the file.

15. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising limiting the hierarchy of sub-components by specifying a pre-determined depth for the hierarchy, wherein decomposing the object comprises decomposing the object to the pre-determined depth.

16. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising limiting the hierarchy of sub-components by defining a property set that identifies individual properties of the object, wherein decomposing the object comprises decomposing the identified individual properties of the object.

17. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising limiting the hierarchy of sub-components by identifying a specified property within the object, wherein decomposing the object comprises decomposing the specified property.

18. (Currently Amended) The computer-readable storage medium of ~~claim 1~~ claim 29, further comprising limiting the hierarchy of sub-components by specifying a pre-determined number that limits the known objects that are serialized into the serialized package by the number.

19. (Currently Amended) At least one computer-readable storage medium having computer executable instructions that provide a method for receiving a package representing a computer-readable object transmitted across a remote boundary, the method comprising:

receiving a serialized package from a remote entity;

identifying a hierarchy of sub-components, the hierarchy representing an object of a first type;

for each sub-component:

identifying a type associated with the sub-component;

determining whether that identified type is within a list of known object types;

~~if the type is identified within a list of known object types,~~
responding to the determining, wherein the responding comprises
instantiating an object of the type and populating at least one property of the object with information obtained from within the serialized package, wherein the instantiating and populating are performed when the identified type is within the list of known object types.

20. (Currently Amended) The computer-readable storage medium of claim 19, wherein the list includes the first type as one of the known object types.

21. (Currently Amended) The computer-readable storage medium of claim 19, wherein the at least one sub-component comprises an unknown object having a type unidentified within the list.

22. (Currently Amended) The computer-readable storage medium of claim 19, wherein a first process on a system receives the serialized package and the remote entity comprises another process on the system.

23. (Currently Amended) The computer-readable storage medium of claim 19, wherein a first process on a system receives the serialized package and the remote entity comprises another process on another system.

24. (Currently Amended) The computer-readable storage medium of claim 19, wherein a first application domain executing within a process receives the serialized package and the remote entity comprises another application domain within the process.

25. (Currently Amended) The computer-readable storage medium of claim 19, wherein the serialized package comprises an XML document.

26. (Currently Amended) A system that communicates objects across a remote boundary, comprising:

a processor;

a memory, the memory being allocated for a plurality of computer-executable instructions which are loaded into the memory for execution by the processor, the computer-executable instructions providing a method for communicating objects across the remote boundary, the method comprising:

~~decomposing an object of a first type into an hierarchy of at least one sub-component based on a list of known object types, each sub-component comprising a known object associated with one of the known object types;~~

~~serializing the known objects into a serialized package; and~~

without relying on a pre-defined definition of a specified data type of an object, decomposing the object into multiple sub-components, wherein the decomposing comprises extracting discernable properties and values for each sub-component;

without relying on the pre-defined definition of the specified data type of the object, serializing the multiple sub-components and their discernable properties and values into a serialized package;
and

transmitting the serialized package to a remote entity,

wherein the decomposing, serializing, and transmitting facilitates communicating objects across the remote boundary.

27. (ORIGINAL) The system of claim 26, wherein a first process on a system transmits the serialized package and the remote entity comprises another process on the system.

28. (Cancelled)

---- NEW ----

29. (NEW) The computer-readable storage medium of claim 1, wherein the decomposing act further comprises:

dividing the multiple sub-components into a hierarchy based upon a list of known object types, the known object types being a type known by the remote entity;

associating an object with one of the known object types, so that object is a known object.

30. (NEW) The computer-readable storage medium of claim 29, wherein just the known objects are serialized in the serializing act.

31. (NEW) The computer-readable storage medium of claim 29, wherein the method further comprises negotiating with the remote entity to determine which object types are known by the remote entity.

32. (NEW) The computer-readable storage medium of claim 1, wherein the method further comprises negotiating with the remote entity to determine which object types are known by the remote entity.